

2.Bölüm

DEĞİŞKEN TIPLERİ

Char : Char tipi, özellikle karakterlerin saklanması içindir.

'a' '1' '=' '&' '\ 'B' ... geçerli karakterlerdir.

Her bir karakter iki üstten virgül arasındadır.

ÖRNEK

```
# include <stdio.h>
main( )
{
char a;
a='z';
printf("a'nın değeri %c\n", a);
}
```

NOT: Bir char değişkeni, bir int değişkeninden daha küçüktür, bu nedenle, bir int değişkeni gibi sayılar alabilmesine karşın, bu sayılar daha küçük olmalıdır. Char tipi bir değişkene, yalnızca 0 ile 255 arasındaki sayılar saklanabilir.

ÖRNEK

```
# include <stdio.h>
main( )
{
char a;
a= 8;
printf("a'nın değeri %c\n", a);
}
```

NOT : Bir char değişkenine 255'den büyük bir sayı sığdırmaya çalışılırsa sonuç 0 olur. Herhangi bir hata oluşmaz, bu C'nin bir esnekliğidir.

FLOAT : Float tipi bir değişken, ondalıklı bir sayı içerir.

8.7 0.898 4.23E9 6.666 ... gibi

Yukarıdaki değerlerden hiç birisi, bir int değişkeninde saklanamaz. Bir int değişkenini 3.88 ile eşitlerseniz sonuç 3 olur.

7 98 330030 gibi tam sayı değişkenleri float değişkenine eşitlenirse 7 7.0, 98 98.0 olarak değiştirilir.

ÖRNEK

```
# include <stdio.h>
main( )
{
float a;
a= 8.909;
printf("a'nın deęeri %f\n", a);
}
```

float deęişkenleri $3.4E-38$ ile $3.4E+38$ arasındaki herhangi bir deęere eşitlenebilirler.

LONG : (uzun tamsayı), yalnızca çok uzun tamsayılar gerektięi zaman kullanılırlar.

-2,147,843 ile 2,147,843 arasında deęer alabilir. Tamsayılar ile yapılan işlemler tam kesinlikte olacaktır. Toplama ve çarpma gibi işlemler. Float tipi, gezer nokta derişkenleri için durum böyle deęildir. Böyle deęişkenler için yalnızca belirli bir sayıda kesinlik rakamı vardır.

DOUBLE : Float tipi deęişkenlerde kesinlik önemli ise bu tür tanımlama yapılır.

TİP ALT ÜST YAKLAŞIK KESİNLİK

Float $3.4E-38$ $3.4E+38$ 7 rakam

Double $1.7E-308$ $1.7E+308$ 14 rakam

Teorik olarak, sonsuz sayıda gezer_nokta sayısı vardır.

.5 4.55 4.555 4.5555 4.55555 4.555555555 4.5555555555555555

Yukarıdaki örnekte son iki sayı float tipi deęişkene eşitlenirse, 4.555555556 olarak yuvarlanır.

Double kullanarak daha iyi bir sonuç elde edilebilir. Sonuç 4.555555555555556 olarak deęişir. Kesinlik önemli ise double kullanılmalıdır.

MAKROLAR : Bir C makrosu, belli bir deęeri bir isimle deęiştirmek için kullanışlı bir yöntem sağlar. Sık kullanılan deęerlere bir isim vermenin kolay bir yöntemidir. Bir C tümcüsü deęildir, derleyici komutuna benzer. Sintaksı (kullanım şekli) şöyledir.

```
# define isim deęer
```

Bu derleyiciye, ne zaman isim görülürse onun yerine belli bir deęer koymasını söyler.

ÖRNEK

```
# include <stdio.h>
# define PI 3.1415 /* PI için deęer belirlenmesi */
main( )
float b=PI;
float a;
a=PI;
```

```
printf("a'nin deęeri %f\n", a);  
printf("a'nin deęeri %f\né, PI);  
}
```